REMARKS

Claims 26-49 are pending in the present application.

Claims 26-49 were rejected.

No claims were amended.

Claims 26-49 remain in the present application.

Reconsideration of the claims is respectfully requested in light of the following arguments, which are made in order to more particularly define the issues for appeal.

In Sections 8-20 of the January 4, 2005 Office Action, the Examiner rejected Claims 26-49 under U.S.C. §103(a) as being unpatentable over U.S. Patent No. 6,006,323 to *Ma et al.* (hereafter, "*Ma*") in view of U.S. Patent No. 5,890,181 to *Selesky et al.* (hereafter, "*Selesky*"). The Applicant respectfully traverses the rejection of Claims 26-49.

The Applicant directs the Examiner's attention to Claim 26, which contains the following unique and novel limitations:

> 26. A method for detecting corruption associated with a stack in a storage device, the method comprising the steps of:
> storing a first predetermined value in a first address location immediately preceding the starting location of the stack;
> detecting the occurrence of a stack operation within the stack; and
> comparing the value in the first address location to the first predetermined value to determine if the stack operation corrupted the first predetermined value stored in the first address location. (*emphasis added*)

The Applicant respectfully asserts that the above-emphasized limitations are not disclosed, suggested, or even hinted at in the *Ma* reference or the *Selesky* reference, or in the combination of the *Ma* and *Selesky* references.

First, regarding the limitation of "[a] method for detecting corruption associated with a stack." In response to the Amendment filed August 9, 2004, the Examiner asserted in Section 4 of the January 4, 2005 Office Action that the *Ma* reference "discloses a method for detecting corruption [i.e., detect over/underflow]," citing Figure 5, the Abstract, and column 4, lines 10-19. The Applicant respectfully submits that the Examiner has misunderstood the teaching of the *Ma* reference.

In fact, the *Ma* reference describes a stack management unit that prevents stack corruption, rather than one that detects corruption associated with a stack, as recited in Claim 26. The *Ma* reference discloses determining a condition that indicates the potential for corruption of the stack (i.e., the overflow condition) and preventing that corruption by moving adequate data out of a primary stack into a secondary stack to relieve the overflow condition. *See Abstract, last sentence; col. 9, lines 25-36.* If the detection of the overflow condition by the *Ma* system were the detection of corruption, as argued by the Examiner, then the actions of the *Ma* system in response – copying primary stack data into the secondary stack – would result in the copying of corrupted stack data into the secondary stack.

Second, regarding the limitation of "comparing [a] value in [a] first address location to [a] first predetermined value." In response to the Amendment filed August 9, 2004, the Examiner

asserted in Section 6 of the January 4, 2005 Office Action that the *Ma* reference discloses comparing the value in a first address location with a predetermined value, citing column 14, lines 58-62. The Examiner also cited column 13, lines 13-22, of the *Ma* reference as disclosing this limitation. The Applicant respectfully submits that the Examiner has misunderstood the language of the claim.

The claim recites a stack in a storage device and a first location, also in the storage device, immediately preceding the start of the stack. The claim further recites examining a value stored in that first location in order to detect corruption associated with the stack. In contrast, the *Ma* reference teaches comparing the locations (or addresses) of first and last elements stored in a stack for the purpose of sensing an overflow condition and preventing stack corruption.

The *Ma* reference clearly describes this functionality at column 8, line 65, through column 9, line 21. "A top of stack (TOS) register 55 . . . indicates the last register written to in the primary stack (e.g., the end of a data element). A bottom of stack (BOS) register 56 indicates the beginning of the oldest element . . . ." *Ma, col. 8, line 65, through col. 9, line 2.* That is, the TOS value is the address of the last used register in the stack, and the BOS is the address of the first used register in the stack. "To calculate the current usage of the primary stack, the TOS value is subtracted from the BOS value by subtractor 57. As shown, the K [least significant bits] of the result are compared to overflow and underflow limitation values . . . ." *Ma, col. 9, lines 16-19.* That is, the address of the register at the top of the stack is subtracted from the address of the register at the bottom of the stack to determine how many stack registers are currently in use. The difference in address values (not the

value stored at a location adjacent the stack, as recited in Claim 26) is the compared to predefined values to sense the overflow condition and thereby prevent stack corruption.

In summary, the *Ma* reference does not teach a method for detecting corruption associated with a stack, as recited in Claim 26. Instead, the *Ma* reference teaches a stack manager for preventing stack corruption by sensing when too little free space remains in the stack and transferring stack data to a secondary stack. Furthermore, the *Ma* reference does not teach comparing the value in a first address location with a predetermined value, as recited in Claim 26. Instead, the *Ma* reference teaches comparing the difference between addresses of the first and last used registers in a stack to predetermined values. Thus, the *Ma* reference does not teach the asserted limitations of independent Claim 26. The Applicant submits that the *Selesky* reference does nothing to overcome the shortcomings of the *Ma* reference.

As such, the unique and novel limitations recited in Claim 26 are not disclosed, suggested, or even hinted at in the *Ma* reference or the *Selesky* reference, or in the combination of the *Ma* and *Selesky* references. This being the case, Claim 26 presents patentable subject matter over the *Ma* and *Selesky* references. Also, Claims 27-37 depend from Claim 26 and contain all of the unique and novel limitations recited in Claim 26. Therefore, Claims 27-37 also are patentable over the *Ma* and *Selesky* references.

The Applicant respectfully asserts that independent Claim 38 contains limitations that are analogous to the unique and novel limitations recited in Claim 26. This being the case, independent Claim 38 presents patentable subject matter over the *Ma* and *Selesky* references. Finally, Claims

39-49, which depend from Claim 38, contain all of the unique and novel limitations recited in Claim 38. Therefore, Claims 39-49 also are patentable over the *Ma* and *Selesky* references.

SUMMARY

For the reasons given above, the Applicant respectfully requests reconsideration and allowance of pending claims and that this Application be passed to issue. If any outstanding issues remain, or if the Examiner has any further suggestions for expediting allowance of this Application, the Applicant respectfully invites the Examiner to contact the undersigned at the telephone number indicated below or at ***jmockler@davismunck.com***.

The Commissioner is hereby authorized to charge any additional fees connected with this communication or credit any overpayment to Deposit Account No. 50-0208.

Respectfully submitted,

DAVIS MUNCK, P.C.

Date: 15 Feb. 2005

John T. Mockler
Registration No. 39,775

P.O. Drawer 800889
Dallas, Texas 75380
Phone: (972) 628-3600
Fax: (972) 628-3616
E-mail: *jmockler@davismunck.com*